

[illegible]

- Figure 6**
- Figure 6 displays a series of 10 histograms showing the distribution of the number of nodes per cluster for different values of  $\alpha$ . The x-axis represents the number of nodes per cluster, ranging from 0 to 10. The y-axis represents the frequency or probability density. The distributions are labeled as follows:
- (a)  $\alpha = 0.0$ : A single peak at 1 node.
  - (b)  $\alpha = 0.1$ : A peak at 1 node, slightly broader than (a).
  - (c)  $\alpha = 0.2$ : A peak at 1 node, more spread out.
  - (d)  $\alpha = 0.3$ : A peak at 1 node, further spread.
  - (e)  $\alpha = 0.4$ : A peak at 1 node, even more spread.
  - (f)  $\alpha = 0.5$ : A peak at 1 node, very spread.
  - (g)  $\alpha = 0.6$ : A peak at 1 node, extremely spread.
  - (h)  $\alpha = 0.7$ : A peak at 1 node, highly spread.
  - (i)  $\alpha = 0.8$ : A peak at 1 node, very highly spread.
  - (j)  $\alpha = 0.9$ : A peak at 1 node, extremely highly spread.
- The histograms illustrate how the distribution of nodes per cluster changes as  $\alpha$  increases from 0.0 to 0.9. As  $\alpha$  increases, the distribution becomes increasingly skewed towards larger numbers of nodes per cluster, indicating a higher degree of clustering or aggregation.

5. The method as recited in claim 1 wherein the buffer associated with first table is created after determining that the parent node of the first node within the hierarchical data is associated with the first relational table.

5 6. The method as recited in claim 1 further comprising a second relational table.

7. The method as recited in claim 5 further comprising creating a second buffer corresponding to the second relational table when a third node within the  
10 hierarchal data corresponds to a second relational table.

8. The method as recited in claim 7 further comprising:  
creating a record in the second buffer associated with the second  
relational table when a child node of the third node indicates that the hierarchical  
15 data comprises data associated with a column in the second relational table; and  
copying the record from the second buffer to the second relational table.

9. The method as recited in claim 8 further comprising copying the record  
from the first buffer to the first relational table substantially in parallel to copying  
20 the record from the second buffer to the second relational table.

10001239 410104

10. The method recited in claim 9 wherein the first relational table has a relational relationship with the second relational table.

11. The method as recited in claim 1 wherein the first buffer comprises a disk  
5 file.

12. The method as recited in claim 1 wherein the hierarchical data comprises an XML document.

10 13. A computer-readable medium bearing computer readable instructions for carrying out the method recited in claim 1.

14. A method for loading hierarchical data into at least two relational tables, comprising:

15 receiving a schema describing a relationship of nodes in the hierarchical data to at least one column in each of the at least two relational tables;

mapping the hierarchical data based on the schema and creating records from the hierarchical data from nodes associated identified as data to be stored in the at least one column in each of the at least two relational tables; and

20 streaming the records into the at least two relational tables.

100125-1101  
10101-521001

15. The method as recited in claim 14 further comprising creating a buffer for each of the at least two relational tables wherein the records are stored before being streamed into the at least two relational tables.

5 16. The method as recited in claim 15 wherein the buffer comprises a file.

17. The method as recited in claim 14 wherein the hierarchical data is not size constrained.

10 18. The method as recited in claim 14 wherein the hierarchical data comprises an XML document.

19. The method as recited in claim 14 wherein the schema comprises an XML schema.

15

20. The method as recited in claim 14 wherein the at least two tables have a relational relationship.

21. The method as recited in claim 14 wherein one of the at least one columns  
20 is a key field in one of the at least two tables and foreign key in the other one of the at least two tables, wherein the method further comprises populating records

10004289 110104

associated with the at least two tables with the data associates with the one of the at least one columns.

22. The method as recited in claim 14 further comprising streaming the record  
5 into the at least two relational tables substantially in parallel.

23. The method as recited in claim 14 wherein the schema is received by way of a network connection.

10 24. A computer-readable medium bearing computer readable instructions for carrying out the method recited in claim 14.

25. A system for loading hierarchical data into at least two relational tables, comprising:

15 instructions for receiving a schema describing a relationship of nodes in the hierarchical data to at least one column in each of the at least two relational tables;

instructions for mapping the hierarchical data based on the schema and creating records from the hierarchical data from nodes associated identified as  
20 data to be stored in the at least one column in each of the at least two relational tables; and

instructions for streaming the records into the at least two relational tables.

10001289 10101

26. The system as recited in claim 25 further comprising instructions for creating a buffer for each of the at least two relational tables wherein the records are stored before being streamed into the at least two relational tables.

5

27. The system as recited in claim 26 wherein the buffer comprises a file.

28. The system as recited in claim 25 wherein the hierarchical data comprises an XML document.

10

29. The system as recited in claim 25 wherein the schema comprises an XML schema.

30. The system as recited in claim 25 wherein the at least two tables have a relational relationship.

15

31. The system as recited in claim 25 wherein one of the at least one columns is a key field in one of the at least two tables and foreign key in the other one of the at least two tables, wherein the system further comprises instructions for  
20 populating records associated with the at least two tables with the data associates with the one of the at least one columns.

FOOTNOTES

32. The system as recited in claim 25 further comprising instructions for streaming the record into the at least two relational tables substantially in parallel.

10001289 110101  
10001289 110101